Using eigenvalue methods for V&V

M.V. Umansky (LLNL)

Acknowledgements: D.A. Baver, J.R. Myra (Lodestar Inc.)

Presented at BOUT++ Workshop
LLNL, Livermore, CA
Sep 3-6, 2013

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Security, LLC, Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.
LLNL-PRES-643674

Outline

- Preliminary remarks on V&V
- Eigenvalue edge code 2DX
- Verification examples
- Validation study example
- IDL built-in tools for eigenvalue solution
- Exercises for practice with eigenvalue methods

V&V = Verification and Validation

Validation

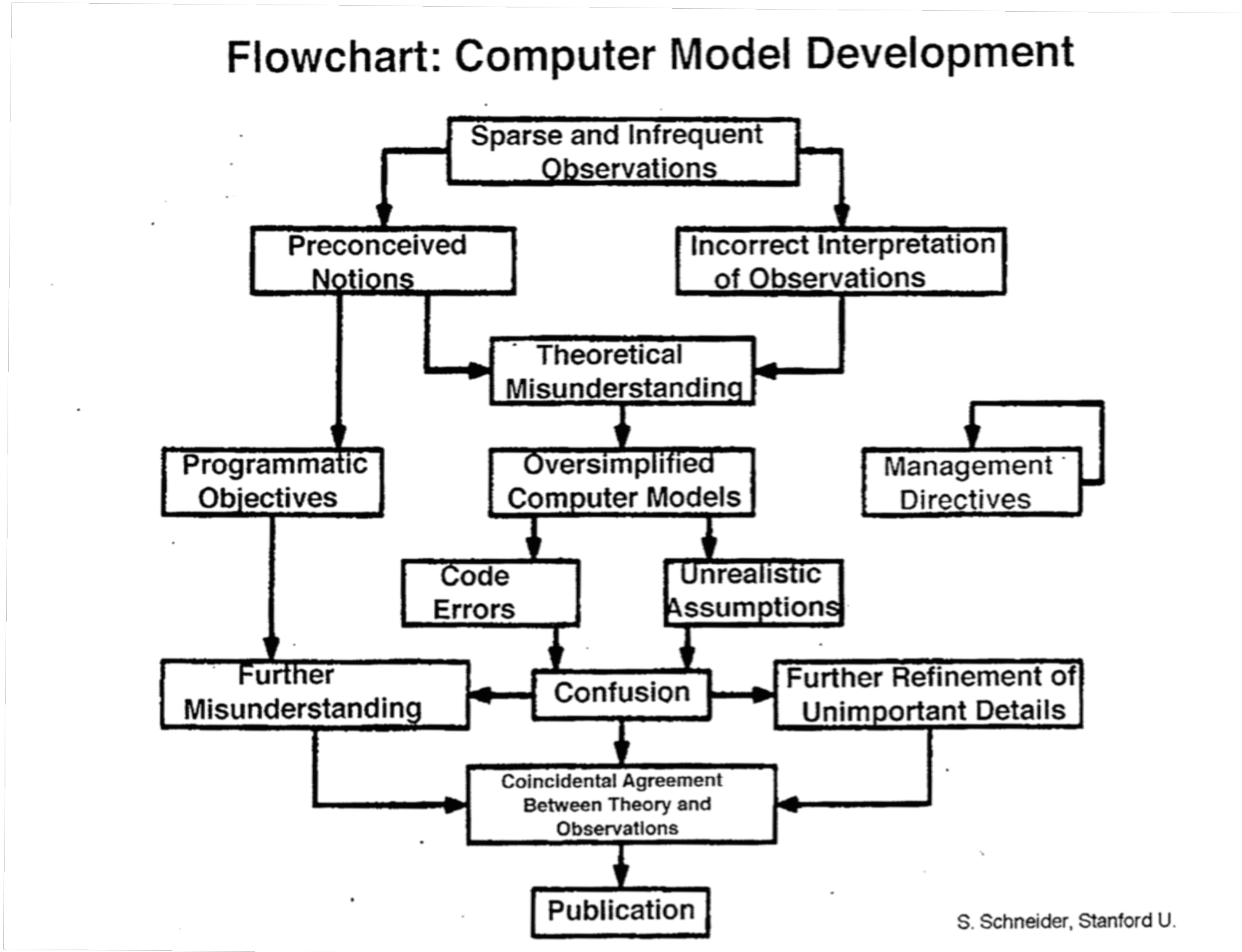
- **Is my model relevant to studied physical phenomena?**
  - **Reliable experimental data?**
  - **Correct analytic theory?**

Verification

- **Is my model implemented correctly?**
  - **Correct numerical procedure?**
  - **No bugs in code implementation?**

In many cases we cannot address validation before we have results from the model so verification comes first

A lot can go wrong in numerical model development…



What can we do to verify a nonlinear time-evolution code like BOUT++?

- **The code implements numerical solution of time-evolution equation which is a large, complicated, nonlinear function**

$$\frac{\partial}{\partial t}\vec{f} = \vec{F}(\vec{f})$$

- **For verification, usual simplification step is linearization**

$$\frac{\partial}{\partial t}\vec{f} = \frac{\partial \vec{F}}{\partial \vec{f}}\vec{f}$$

- **The time-evolution solution is the fastest-growing (or least damped) eigenmode of the matrix M**

$$-i\omega\vec{f} = \hat{M}\,\vec{f}$$

- **Eigenvalue solution is a natural choice for independent cross-check of a time evolution code**

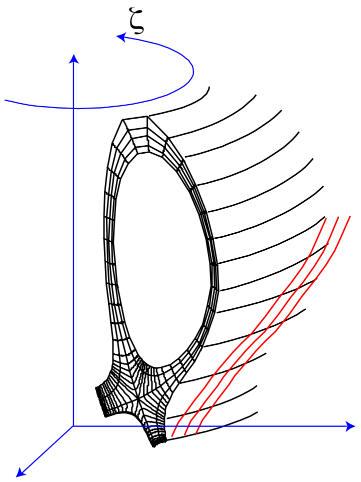
2DX eigenvalue code* provides new capabilities for edge/SOL analysis

- Solves linearized eigenvalue problem in R-Z plane for each toroidal mode number n.
- Inputs actual magnetic divertor geometry for edge and SOL
  - experimental or analytical input profiles of n_e, T_e, T_i and E_r
- Uses a specialized equation parser to input an arbitrary fluid physics model
  - ideal MHD, drift-resistive MHD, full 6-field fluid model etc.
- Sparse matrix package SLEPc enables high resolution
- Originally devised primarily as a linear benchmark tool for nonlinear codes
- Applications for analysis of experiments (QC-mode, edge marginal stability, density limit)
- 2DX is a copyrighted code developed by Lodestar Research Corp. (in collaboration with LLNL)

* D. A. Baver, J. R. Myra, and M.V. Umansky, Comp. Phys. Comm. 182, 1610, (2011)

2DX eigenvalue code* provides new capabilities for edge/SOL analysis

- Transform to field-following coordinates (more efficient for field-aligned functions)



$$\delta\Phi = \delta\Phi_f(\psi,\theta)\exp\left(in\zeta - in\int_{\theta_0}^{\theta} d\theta\,\nu \right)$$

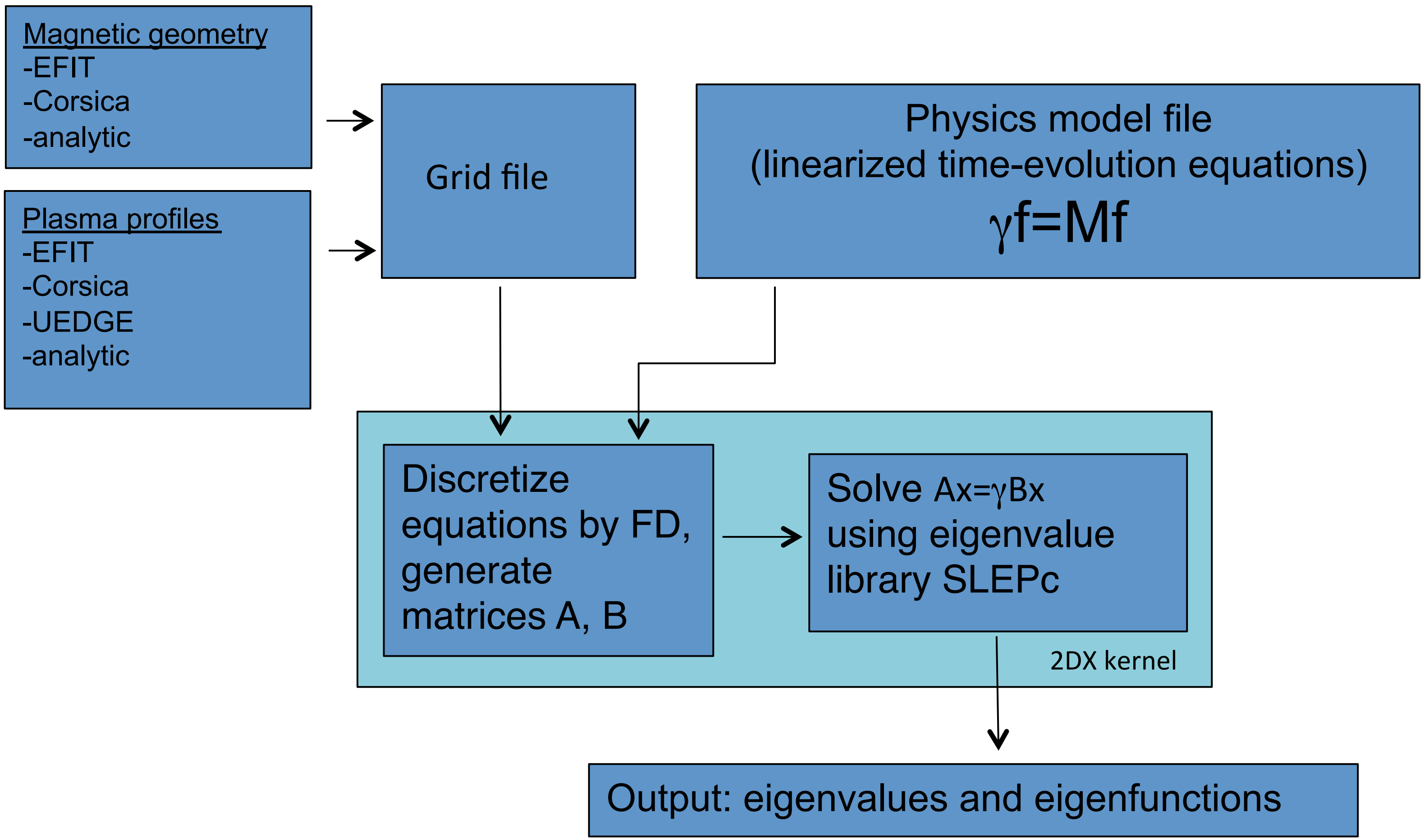
- Numerically solve for the envelope $\delta\Phi_f$

- Periodicity invoked by phase-shift-periodic boundary condition

$$\delta\Phi_f(\psi,2\pi) = \delta\Phi_f(\psi,0)e^{2\pi inq} \qquad q = \frac{1}{2\pi}\int_0^{2\pi} d\theta\,\nu$$

- Equivalent to BOUT/BOUT++ formulation for linear single toroidal mode BUT
  - much more efficient (a few orders of magnitude)
  - allows finding subdominant modes
  - allows finding modes near specific complex ω

2DX eigenvalue code* provides new capabilities for edge/SOL analysis



Large collection of verification test problems has been produced with 2DX for edge plasma community

http://www.lodestar.com/research/vnv/

- A. *Table of benchmark comparisons*
- B. *Resistive Ballooning*
- C. *Resistive Drift*
- D. *Slab Ion Temperature Gradient (ITG)*
- E. *Geodesic Acoustic Mode (GAM)*
- F. *Ideal Kink*
- G. *Parallel Kelvin Helmholtz*
- H. *Toroidal Ion Temperature Gradient (ITG)*
- I. *Edge Localized Mode (ELM)*
- J. *Kinetic Resistive Ballooning*

Each test problem has detailed description, cross-checked against other codes and asymptotic analytic solutions

Large collection of verification test problems has been produced with 2DX for edge plasma community

http://www.lodestar.com/research/vnv/

- A. *Table of benchmark comparisons*
- B. *Resistive Ballooning*
- C. *Resistive Drift*
- D. *Slab Ion Temperature Gradient (ITG)*
- E. *Geodesic Acoustic Mode (GAM)*
- F. *Ideal Kink*
- G. *Parallel Kelvin Helmholtz*
- H. *Toroidal Ion Temperature Gradient (ITG)*
- I. *Edge Localized Mode (ELM)*
- J. *Kinetic Resistive Ballooning*

Each test problem has detailed description, cross-checked against other codes and asymptotic analytic solutions

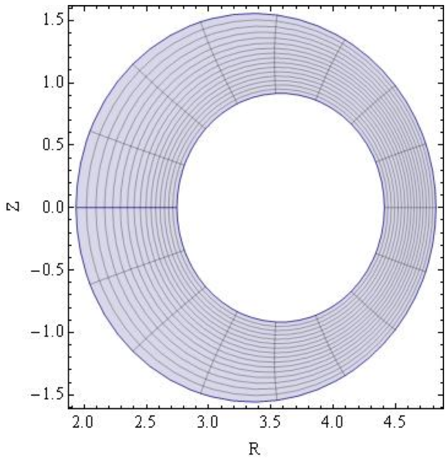
Using 2DX for verification: ELM benchmark

- **Model equations**

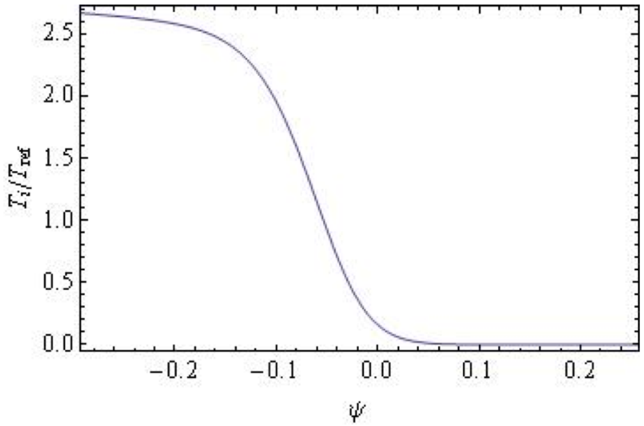
$$\gamma\nabla_\perp^2\delta\phi = \frac{2B}{n}C_r n\delta T_i - \frac{B^2}{n}\partial\nabla_\perp^2\delta A + i\frac{Bk_b}{n}\delta A\partial_r\frac{J_\parallel}{B}$$
$$\gamma\delta T_i = -i\frac{k_b}{B}\delta\phi\partial_r T_i$$
$$\gamma\left(\frac{n}{\delta_{er}^2}\right)\delta A = -n\mu\nabla_\parallel\delta\phi$$

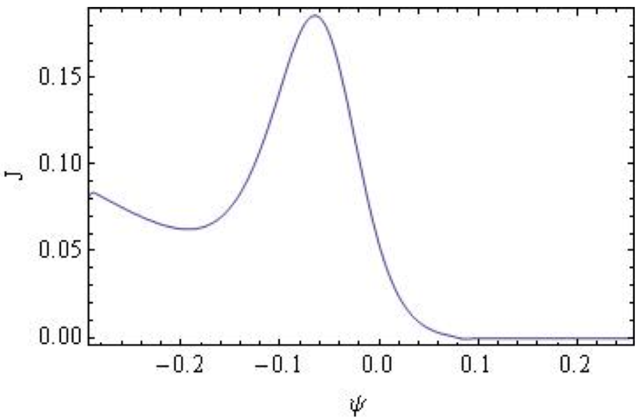
$$C_r = \mathbf{b}\times\kappa\cdot\nabla = -\kappa_g R B_p\partial_x + i(\kappa_n k_b - \kappa_g k_\psi)$$
$$\nabla_\perp^2 = -k_b^2 - \jmath B(k_\psi - i\partial_x R B_p)(1/\jmath B)(k_\psi - iRB_p\partial_x)$$
$$\partial_\parallel Q = B\nabla_\parallel(Q/B)$$
$$\nabla_\parallel = \jmath\partial_y$$

- **Geometry description**



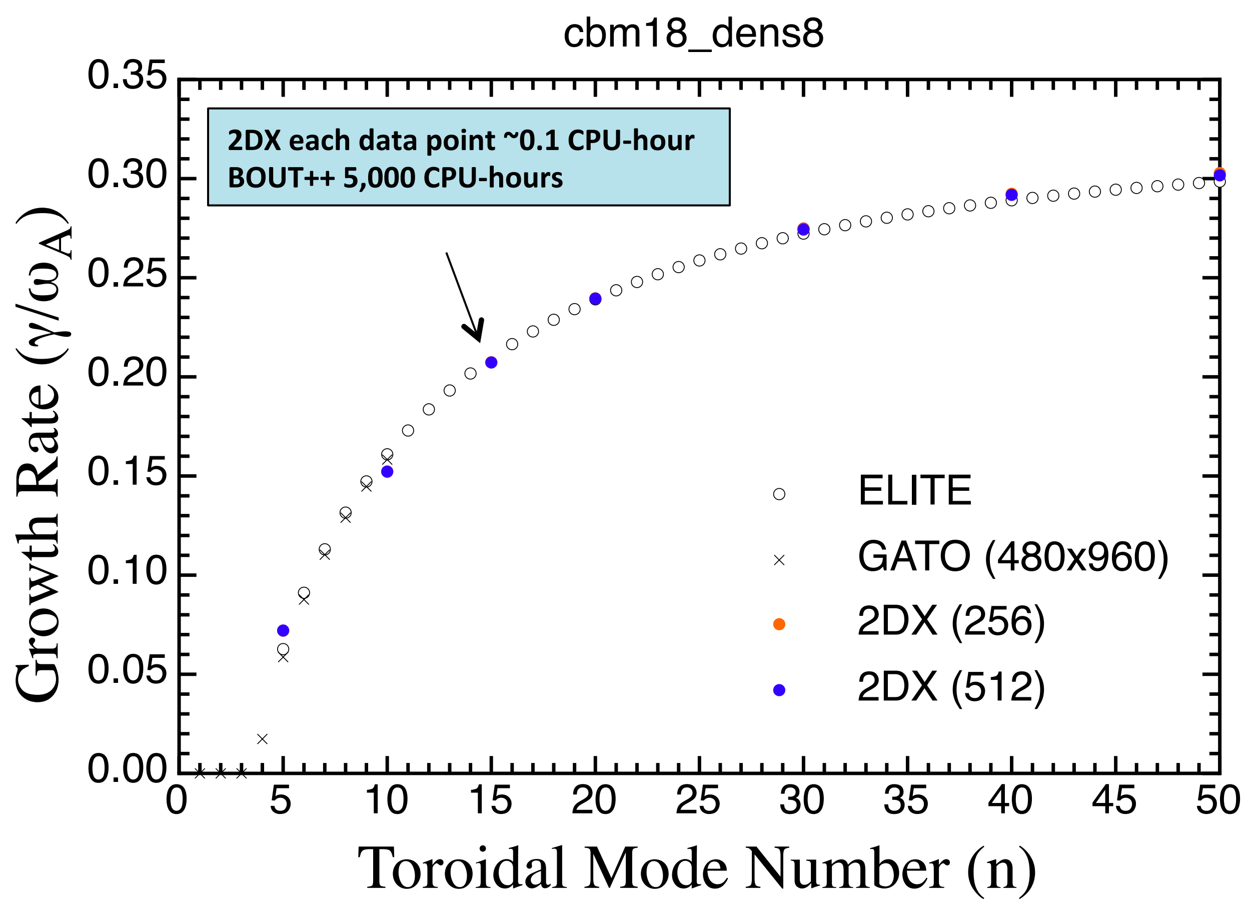
- **Plasma profiles**





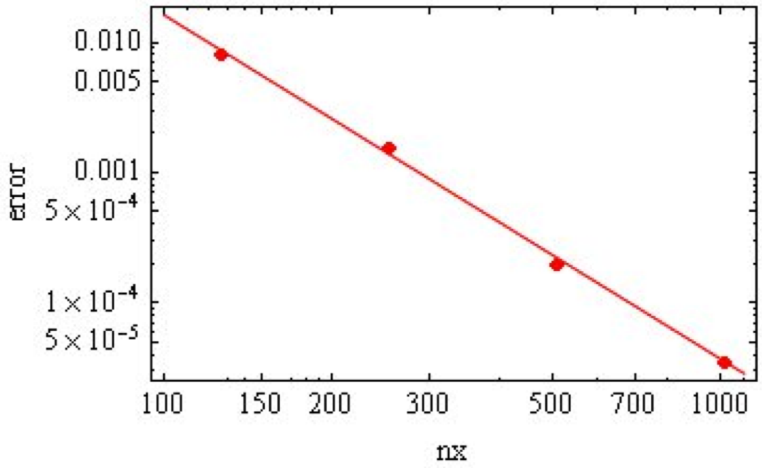
Using 2DX for verification: ELM benchmark

- **Dispersion relation (plots and tables)**

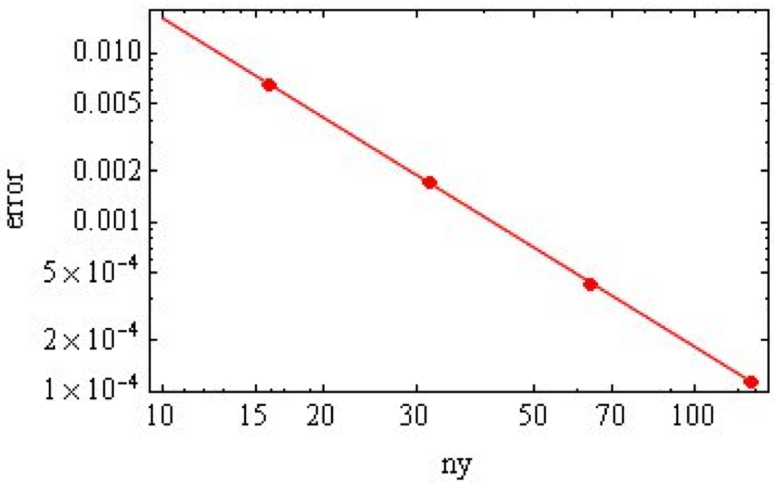


Using 2DX for verification: ELM benchmark

- **Grid convergence studies results (plots and tables)**



n	$Re(\gamma)/\omega_A$ 2DX (nx=512)	$Re(\gamma)/\omega_A$ 2DX (nx=256)	γ/ω_A BOUT++ (nx=512)	γ/ω_A BOUT++ (nx=256)
5	.0865383	.0864301	.0894333	.0916790
10	.182753	.182911	.194466	.195453
15	.248903	.249047	.250183	.254074
20	.287606	.287774	.292470	.295150
30	.329408	.329749	.330025	.348405
40	.350433	.351111	.361835	.383246
50	.362255	.363654	.386312	.403599
60	.369492	.372193	.398018	.426154
70	.374249	.378379	.410997	.443173
80	.377585	.381714	.419887	.453644
90	.380091	.385232	.427058	.460371
100	.382127	.386651	.430449	.470444

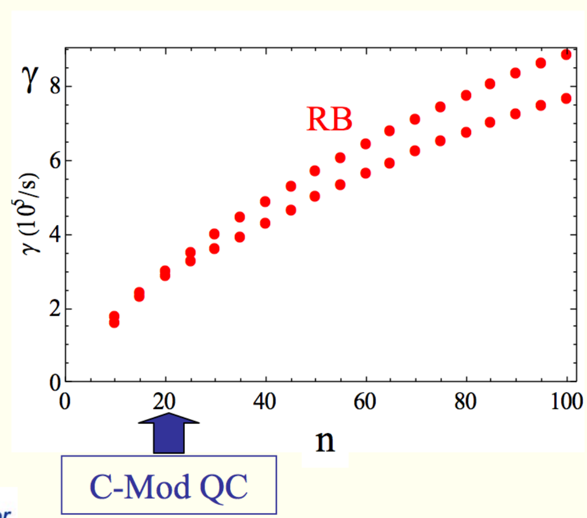


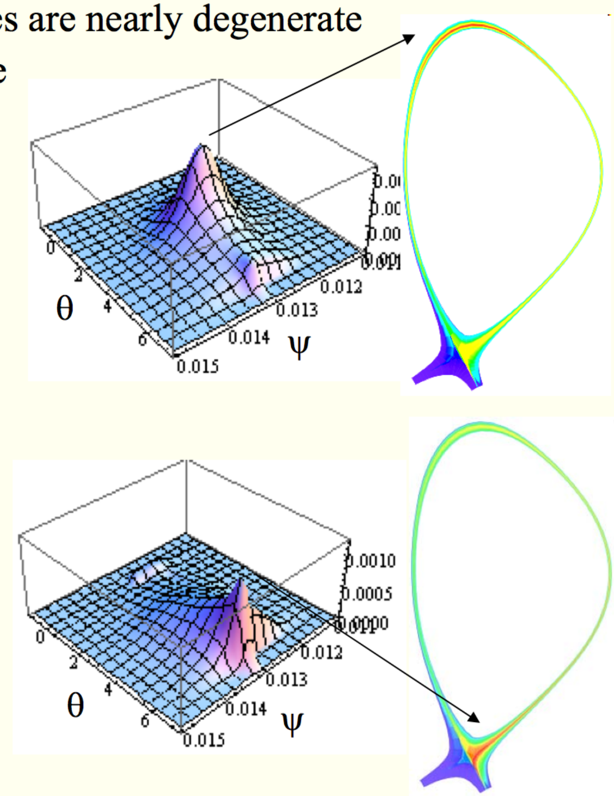
Using 2DX for validation: Understanding linear stability of C-Mod EDA plasma

From Myra et al, APS 2011 talk

X-pt geometry supports two RB branches

- 2 fastest resistive ballooning branches are nearly degenerate
- n = 20 mode is in experimental range
  - C-Mod: n ~ 17 to 21
  - spatially confined by X-pts (outboard side)
- BUT no spectral peak near n = 20





Alcator C-Mod

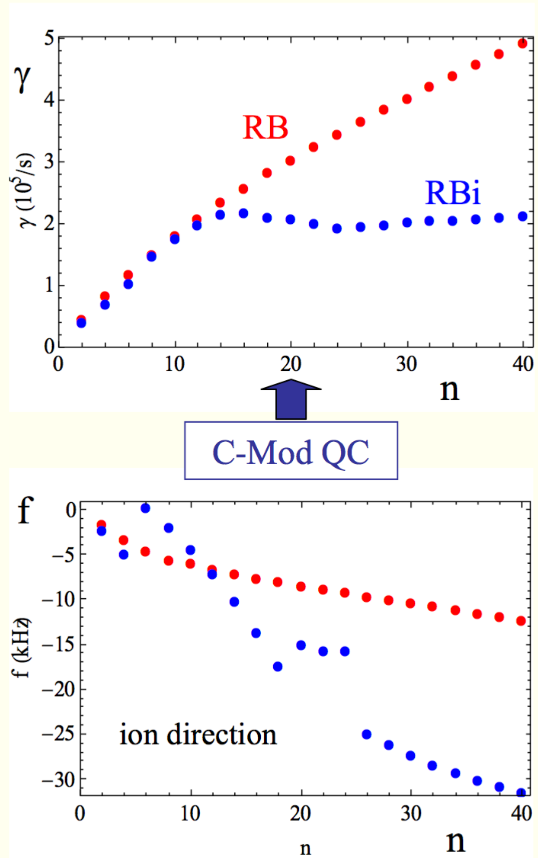
Lodestar

Using 2DX for validation: Understanding linear stability of C-Mod EDA plasma

From Myra et al, APS 2011 talk

Ion diamagnetism plateaus growth rates for n > 15

- BUT still no strong spectral peak near n = 20
  - strength of peak is somewhat sensitive to n vs. T_i profile (including SOL)
- $f_{n=20} \sim -15$ kHz (ion direction) in plasma frame
- only fastest modes shown



- C-Mod data:
  - f ~ 55 to 70 kHz, e-direction (lab frame), possibly i-direction (plasma frame)
  - n ~ 17 to 21

Alcator C-Mod

Lodestar

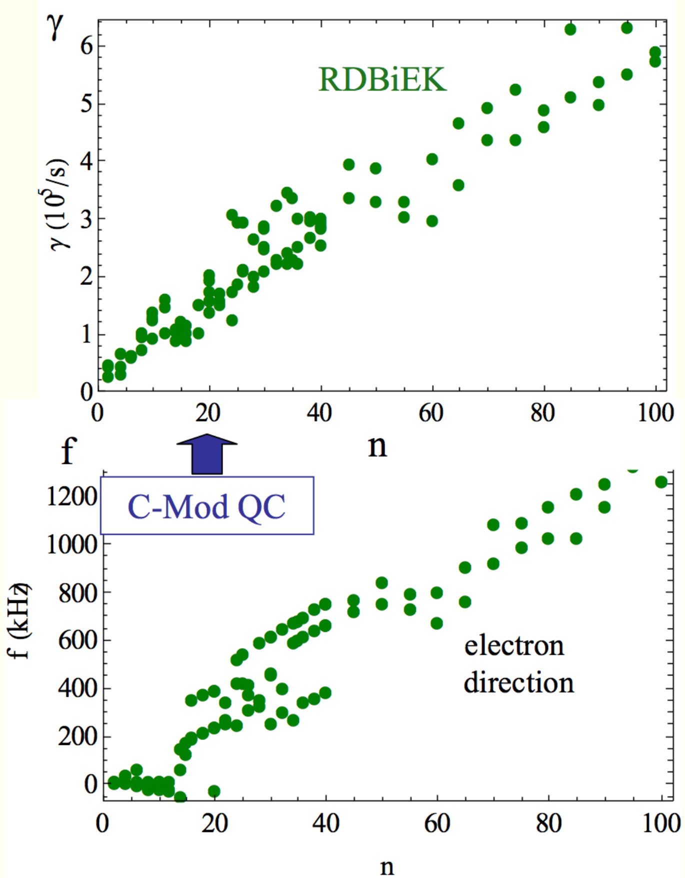
Fusion Energy Sciences

Using 2DX for validation: Understanding linear stability of C-Mod EDA plasma

From Myra et al, APS 2011 talk

Drift wave physics: complex spectrum with $\gamma \propto n$

- RDBiEK = more complete model including resistive ballooning, ion diamagnetism, drift waves, sheared Er, and KH physics
- 2 fastest modes shown
- No spectral peak near n = 20
- $f_{n=20}$ ~ 200+ kHz (electron direction) in lab frame
  - probably sensitive to profile uncertainties



No linear modes with strong peak growth rates in the relevant range of wave-numbers have emerged yet.

Alcator C-Mod *Lodestar*

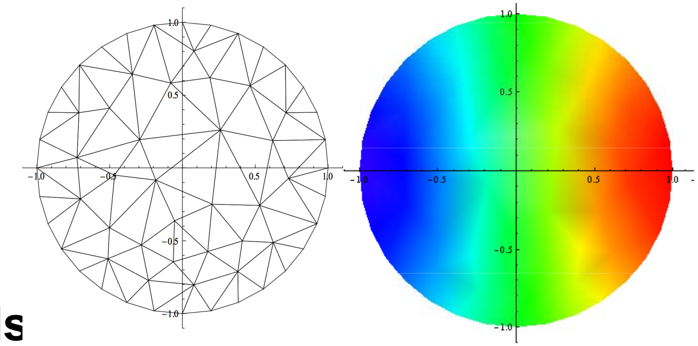
Fusion Energy Sciences

Ongoing development of new capabilities 2DX → Arbiter

- Arbitrary Topology Equation Reader (ArbiTER)

- Target applications areas
  - Full linear kinetic or gyrokinetic models.
  - Non-axisymmetric geometries (e.g., stellarators)
  - Topologies beyond simple X-point (e.g., snowflake divertors)

- New capabilities (under development)
  - Multiple dimensions
  - Arbitrary connectivity
  - Variable number of dimensions
  - Finite elements and unstructured grids
  - Integrated parallelization



Using IDL built-in tools for eigenvalue analysis
Case study: resistive drift instability

Linearized equations for n_i, $v_{||e}$, Φ in cylindrical annular geometry ➔ **Nonlinear eigenvalue problem for $\varphi(r)$**

$$\partial_t N + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla N = -\mathbf{b_0} \times \nabla_\perp \phi \cdot \nabla N_0 - N_0 \nabla_\parallel v_{\parallel e},$$

$$\partial_t v_{\parallel e} + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla v_{\parallel e} = -\mu \frac{T_{e0}}{N_0} \nabla_\parallel N + \mu \nabla_\parallel \phi - \nu_e v_{\parallel e}, \quad \text{(A3)}$$

$$\begin{aligned} N_0 \nabla_\parallel v_{\parallel e} = & -\nabla_\perp \cdot (N_0 \partial_t \nabla_\perp \phi + \partial_t N \nabla_\perp \phi_0 \\ & + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla(N_0 \nabla_\perp \phi_0) \\ & + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla(N_0 \nabla_\perp \phi) \\ & + \mathbf{b_0} \times \nabla_\perp \phi \cdot \nabla(N_0 \nabla_\perp \phi_0) \\ & + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla(N \nabla_\perp \phi_0) \\ & + N_0 \nu_{in} \nabla_\perp \phi_0 + N_0 \nu_{in} \nabla_\perp \phi + N \nu_{in} \nabla_\perp \phi_0). \end{aligned}$$

$$\lambda_v(r, \tilde{\omega}) = ik_\parallel \mu \frac{1 - \dfrac{T_{e0}}{\tilde{\omega} L_n} \dfrac{m_\theta}{r}}{\nu_e - i\tilde{\omega} + ik_\parallel^2 \mu \dfrac{T_{e0}}{\tilde{\omega}}}, \qquad L_n = -\frac{N_0}{N_0'}, \qquad \tilde{\omega} = \omega - \frac{m_\theta}{r} \phi_0'.$$

$$\lambda_N(r, \tilde{\omega}) = \frac{ik_\parallel^2 \mu + \dfrac{m_\theta}{r} \dfrac{1}{L_n} (\nu_e - i\tilde{\omega})}{\tilde{\omega}(\nu_e - i\tilde{\omega}) + ik_\parallel^2 \mu T_{e0}},$$

$$C_2(r,\omega)\varphi'' + C_1(r,\omega)\varphi' + C_0(r,\omega)\varphi = 0$$

$$C_2(r)\phi'' + C_1(r)\phi' + C_0(r)\phi = 0, \quad \text{(A4)}$$

$$C_2(r) = (\nu_{in} - i\tilde{\omega}), \quad \text{(A5)}$$

$$C_1(r) = (\nu_{in} - i\tilde{\omega})\left(\frac{1}{r} - \frac{1}{L_n} + \phi_0' \lambda_N\right) + im_\theta \frac{1}{rL_n} \phi_0', \quad \text{(A6)}$$

$$C_0(r) = (\nu_{in} - i\tilde{\omega})\left[-\frac{m_\theta^2}{r^2} + \lambda_N \phi_0' \left(\frac{1}{r} - \frac{1}{L_n}\right) + (\lambda_N \phi_0')'\right] \quad \text{(A7)}$$

$$+ \frac{im_\theta}{r^3}\left[\phi_0' - r\phi_0'' - r^2 \phi_0''' - \frac{r}{N_0}(rN_0'\phi_0')' + \frac{r^2}{L_n}\phi_0''\right] \quad \text{(A8)}$$

$$+ ik_\parallel \lambda_v + im_\theta \frac{1}{r} \lambda_N \phi_0' \phi_0'', \quad \text{(A9)}$$

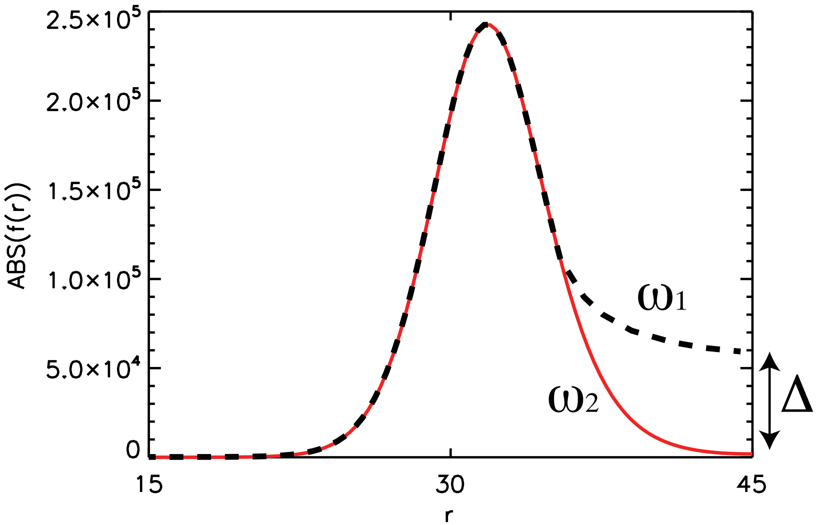
Can be solved by the shooting method

From Popovich et al., Phys. Plasmas 17, 102107, 2010

Using IDL built-in tools for eigenvalue analysis
Case study: resistive drift instability

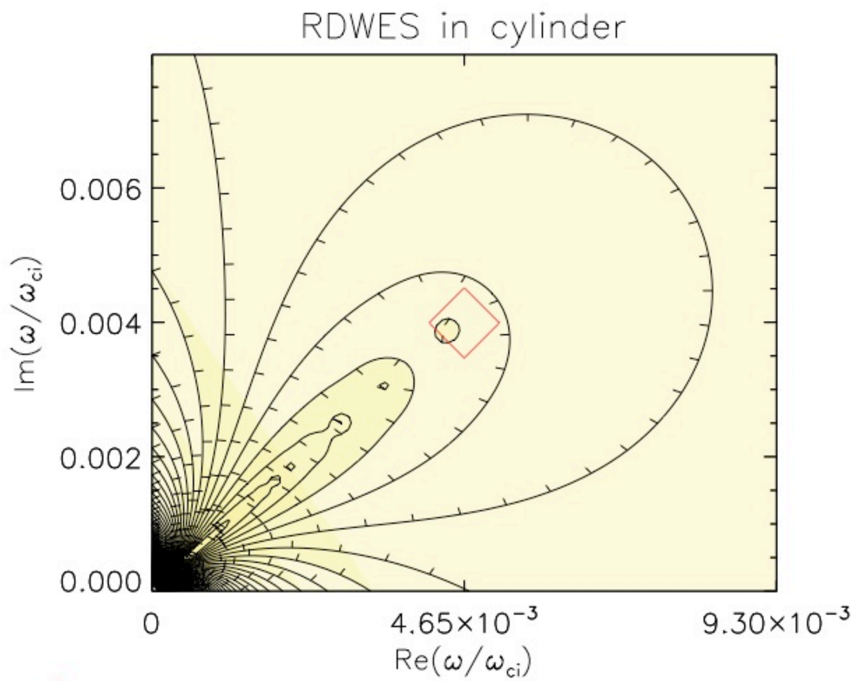
Nonlinear eigenvalue problem for φ(r)

$$C_2(r,\omega)\varphi'' + C_1(r,\omega)\varphi' + C_0(r,\omega)\varphi = 0$$



Shooting method in a action

Scan of residual Δ in complex ω plane



Using IDL built-in tools for eigenvalue analysis
Case study: resistive drift instability

Linearized equations for n_i, $v_{||e}$, Φ in cylindrical annular geometry ➔ **~~Nonlinear eigenvalue problem for φ(r)~~**

$$\cancel{C_2(r,\omega)\varphi'' + C_1(r,\omega)\varphi' + C_0(r,\omega)\varphi = 0}$$

$$\partial_t N + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla N = -\mathbf{b_0} \times \nabla_\perp \phi \cdot \nabla N_0 - N_0 \nabla_\| v_{\|e},$$

$$\partial_t v_{\|e} + \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla v_{\|e} = -\mu \frac{T_{e0}}{N_0} \nabla_\| N + \mu \nabla_\| \phi - \nu_e v_{\|e}, \tag{A3}$$

$$\begin{aligned} N_0 \nabla_\| v_{\|e} = -\nabla_\perp \cdot (&N_0 \partial_t \nabla_\perp \phi + \partial_t N \nabla_\perp \phi_0 \\ &+ \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla (N_0 \nabla_\perp \phi_0) \\ &+ \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla (N_0 \nabla_\perp \phi) \\ &+ \mathbf{b_0} \times \nabla_\perp \phi \cdot \nabla (N_0 \nabla_\perp \phi_0) \\ &+ \mathbf{b_0} \times \nabla_\perp \phi_0 \cdot \nabla (N \nabla_\perp \phi_0) \\ &+ N_0 \nu_{in} \nabla_\perp \phi_0 + N_0 \nu_{in} \nabla_\perp \phi + N \nu_{in} \nabla_\perp \phi_0). \end{aligned}$$

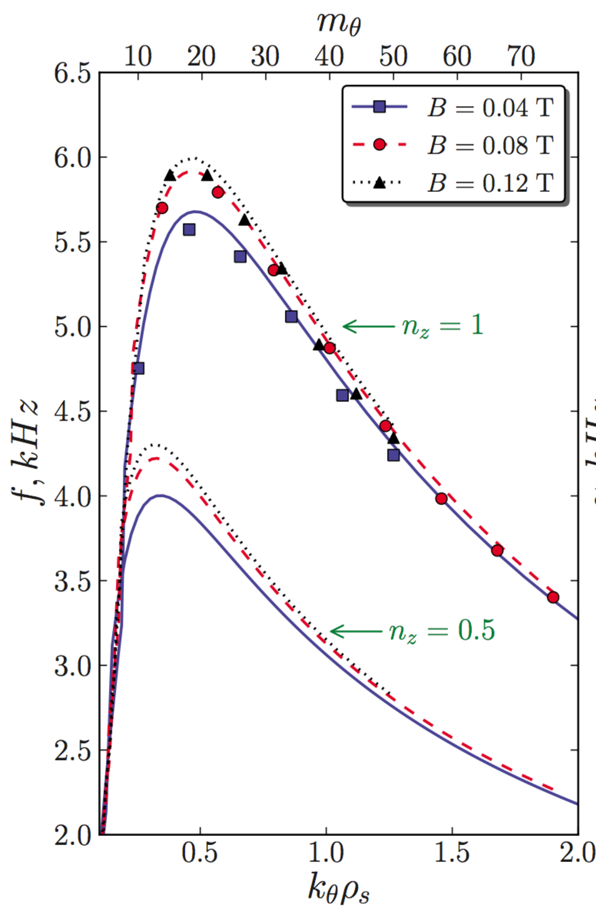
Solve it as a 3xN size linear algebra eigenvalue problem

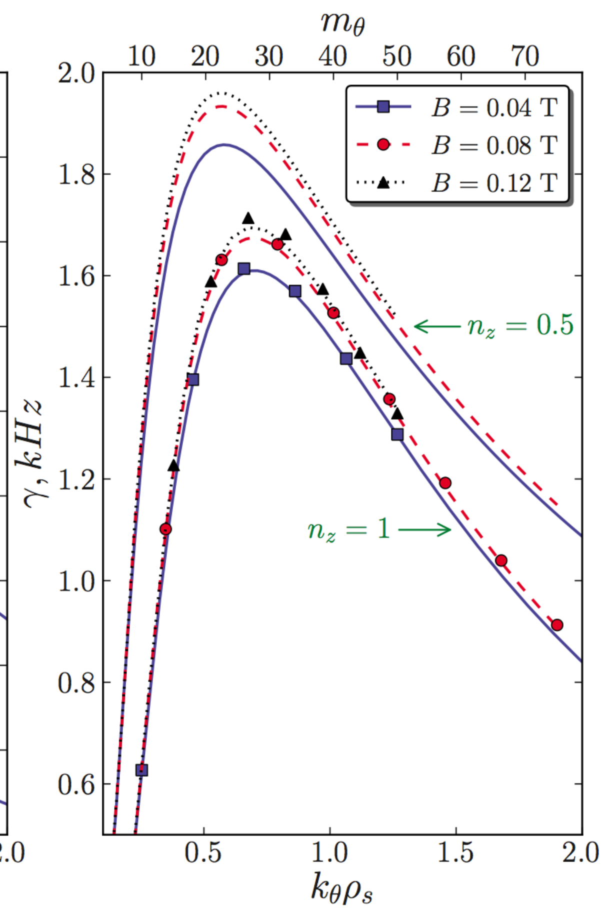
-a bit more programming work

BUT

-much more robust procedure
-can use standard packages

Eigenvalue solution is used routinely to verify BOUT++ time-evolution results





From Popovich et al., Phys. Plasmas 17, 102107, 2010

Suggested exercises for practice with eigenvalue solution

- On hopper do: more ~umansky/BOUT_Workshop_2013/readme

- Copy to your area on hopper and experiment with
  - shooting method example
  - built-in eigenvalue solver example
- Experiment with the codes using instructions in README files
- Try to understand the code in both examples
- Implement (in IDL, Python, or MATLAB) your own solution and find eigenvalues and eigenmodes for acoustic wave equations using
  - shooting method
  - built-in linear eigenvalue solver routine

$$\begin{cases} \dfrac{\partial n}{\partial t} = -\dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial t} = -\dfrac{\partial n}{\partial x} \end{cases} \qquad x \in [0,1] \qquad n_\Gamma = 0, \quad \left.\dfrac{\partial u}{\partial x}\right|_\Gamma = 0$$